



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/712,771	11/12/2003	Yuval Gilboa	017900-003810US	7406
59734 7590 04/20/2007 TOWNSEND AND TOWNSEND AND CREW, LLP TWO EMBARCADERO CENTER EIGHTH FLOOR SAN FRANCISCO, CA 94111			EXAMINER DASGUPTA, SOUMYA	
			ART UNIT 2109	PAPER NUMBER
SHORTENED STATUTORY PERIOD OF RESPONSE		MAIL DATE	DELIVERY MODE	
3 MONTHS		04/20/2007	PAPER	

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

<b>Office Action Summary</b>	Application No. 10/712,771	Applicant(s) GILBOA, YUVAL	
	Examiner Soumya (Ronnie) Dasgupta	Art Unit 2109	

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☐ Responsive to communication(s) filed on \_\_\_\_.
- 2a) ☐ This action is **FINAL**.                      2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-30 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
     Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
     Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |  |
|--|--|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. ____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)   | 5) <input type="checkbox"/> Notice of Informal Patent Application                      |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)<br>Paper No(s)/Mail Date <u>7/19/2004</u> | 6) <input type="checkbox"/> Other: ____  |

## DETAILED ACTION

### ***Claim Rejections - 35 USC § 112***

1. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

2. Claim 1-4, 8-25, & 27-30 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

**Claim 1:**

The term "defining a relationship" is vague and indefinite. The applicant fails to explain what the term "relationship" means. The examiner assumes that the applicant means the relationship is defined by Fig. 11 A of this application.

**Claim 2:**

The term "canonic representation" is broad and indefinite. The applicant fails to explain what the term "relationship" means. According to [www.dictionary.com](http://www.dictionary.com), "canonic" is defined as

1. pertaining to, established by, or conforming to a canon or canons.
2. included in the canon of the Bible.
3. authorized; recognized; accepted: canonical works.
4. Mathematics. (of an equation, coordinate, etc.) in simplest or standard form.
5. following the pattern of a musical canon.
6. Linguistics. (of a form or pattern) characteristic, general or basic: the canonical form of the past tense; a canonical syllable pattern.
- noun 7. canonicals, garments prescribed by canon law for clergy when officiating.

The examiner assumes the term "canonic representation" to mean source code.

**Claim 3:** (see Claim 2 for "canonic representation")

**Claim 4:** (see Claim 1 for "relationship")

**Claim 8:**

Art Unit: 2109

The term "enterprise portal" is vague and indefinite. However, the examiner assumes the meaning of enterprise portal "as one of the most popular ways in which enterprises can allow their employees and customers to search and access corporate information [; it] is a single gateway for users, such as employees, customers and company's partners to log into and retrieve corporate information, company history and other services or resources" as defined by Wikipedia at [http://en.wikipedia.org/wiki/Enterprise\\_portal](http://en.wikipedia.org/wiki/Enterprise_portal).

Claim 9: (see Claim 1 for "relationship")

Claim 10:

The term "business function" is vague and indefinite. The examiner assumes the definition of "business function" to mean an object oriented coded function or procedure that is related to business applications.

Claim 11: (see Claim 2 for "canonic representation")

Claim 12: (see Claim 10 for "business functions")

Claim 13: (see Claim 10 for "business functions")

Claim 14: (see Claim 10 for "business functions")

Claim 15: (see Claim 2 for "canonic representation")

Claim 16: (see Claim 2 for "canonic representation")

Claim 17: (see Claim 2 for "canonic representation")

Claim 18: (see Claim 10 for "business functions")

Claim 19: (see Claim 2 for "canonic representation")

Also, the term "verification" is vague and indefinite. The examiner interprets "verification" to mean ascertaining that the source code does not get altered.

Claim 20:

The term "performance tuning" is stated, but not described in the specifications. The examiner interprets "performance tuning" to be optimization and self-correction.

Claim 21: (see Claim 10 for "business functions")

Claim 22: (see Claim 10 for "business functions")

Also, the term "reusable application components" is vague and indefinite. The examiner assumes the definition of "reusable application components" to mean a computer application or its components that can be used over and over again. This can be any computer application, (i.e. MS Word) or any function of the

Art Unit: 2109

application (i.e. copy and paste command) that can be used extensively without limitations.

Claim 23: (see Claim 22 for “reusable application components”)

Claim 24: (see Claim 10 for “business functions”)

Also, the term “distributed computer system” is vague and indefinite. The examiner uses the definition of distributed system as defined by Microsoft Computer Dictionary as a “non centralized network consisting of numerous computers that can communicate with one another and that appear to users as parts of a single, large, accessible ‘storehouse’ of shared hardware, software, and data. Therefore, the examiner concludes that a “distributed computer system” is a type of parallel processing network system ([http://en.wikipedia.org/wiki/Distributed\\_computing](http://en.wikipedia.org/wiki/Distributed_computing)).

Claim 25: (see Claim 2 for “canonic representation”)

Claim 27: (see Claim 10 for “business functions”)

Claim 28: (see Claim 10 for “business functions”)

Claim 29: (see Claim 2 for “canonic representation”)

Claim 30: (see Claim 10 for “business functions”)

### ***Claim Rejections - 35 USC § 102***

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claim 1-8, 9-19, 21-22, & 28-30 are rejected under 35 U.S.C. 102(b) as being anticipated by Microsoft Access 2000 (for hereon will be known as Access).

Art Unit: 2109

## Claim 1:

Access teaches a method for generating a user interface, the user interface being configured for use in a client-server environment (pg 242 for networks), the method comprising: providing an editor for designing a visual representation of a user interface, the editor providing a workspace and a task panel to be displayed on a display device on a client system, the workspace being provided to design the visual representation thereon, the task panel providing a plurality of elements for use in designing the visual representation, one or more of the elements being associated with a server system remotely located from the client system (pg 242 for networks; the examiner notes that Access is a GUI editor); selecting a first actor from the task panel, the first actor being a data source object that is one of the elements and includes application logic needed to access application layer provided in the server system ; inserting the first actor into the workspace (pg 224-230 shows inserting an operator into a workspace by using a toolbar); inserting a second actor selected from the task panel into the workspace (pg 224-230 shows inserting an operator into a workspace by using a toolbar); diagrammatically defining a relationship between the first actor and the second actor (pg 165 – relationship tool bar) and generating executable code from the first and second actors and the relationship thereof. (pg 21-43 for Visual Basic code) The examiner notes that it is well known in the art that Visual Basic is a form of source code that can be can be compiled into executable code, like any

other source code. See Larry Ayers article on how source code can be compiled into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 2:

Access teaches a generating a canonic representation of the first and second actor and the relationship thereof, wherein the executable code is generated from the canonic representation. (pg 21-43 for Visual Basic code) The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be compiled into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 3:

Access teaches a method wherein the generated executable code is compatible with a first platform, wherein the canonic representation is used to generate executable code for a second platform. (pg 21-43 for Visual Basic code) The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be compiled into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 4:

Access teaches a method comprising: inserting an operator to the workspace, the operator being configured to process data in a specific way (pg 224-230 shows inserting an operator into a workspace by using a toolbar); diagrammatically defining a relationship between the second actor and the operator (pg 165 – relationship tool bar).

Claim 5:

Access teaches a method comprising: storing an identifier of the first actor in a work session associated with the current instance of the editor, wherein the identifier of the first actor is used to call the first actor stored in the server system during a runtime to have the first actor perform a predetermined task (pg 21-43 for Visual Basic source codes and pg 207 for operator function call). The examiner interprets this process a simple operator function call.

Claim 6:

Access teaches a method comprising: logging on to the server system to launch the editor (pg 242). The examiner interprets “server system” to be a network system. Access allows a user to login to a network in order to gain admittance to use the MS Access from a client server.

Claim 7:



Access teaches storing the generated executable code to a repository in the server system. (pg 242 for network system and pg 19, 31, and 37 on storing)

Claim 9:

A method for generating a user interface using a modeling system, comprising: providing an editor for designing a visual representation of a user interface from a server system to a client system, the editor providing a workspace and a task panel to be displayed on a display device on the client system, the workspace being provided to design the visual representation thereon, the task panel providing a plurality of elements for use in designing the visual representation, one or more of the elements being associated with the server system remotely located from the client system (pg 242 for networks; the examiner notes that Access is a GUI editor); displaying a scenario selected by a user on the workspace, the scenario being compatible with user requirements for the user interface, the scenario including a plurality of interleaved scenes (p. 327 - 333 for creating web pages for applications. The examiner interprets a web page is a type of scenario); defining each of the plurality of scenes according to inputs received from the user, each scene including concurrently active and collaborating actors, the actors being specialized computational units that represent threads of activities, where each scene is defined by diagrammatically defining a relationship between the actors associated with that scene (pg 165 for diagrammatic relationship tool bar); generating a canonic representation of a

Art Unit: 2109

model represented by the scenario and the scenes (pg 21-43 for Visual Basic source codes); and generating executable of from the canonic representation (pg 21-43 for Visual Basic source codes). The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be complied into an executable code (<http://www.linuxjournal.com/article/0216>).

## Claim 10:

Access teaches a method for generating a user interface in a distributed computer system, the method comprising: displaying a first business function component selected by a first user on a first display area of a front-end system, the first business function component being associated with first application logic to access a first business application provided in a server system (pg 242 for networks); displaying a second business function component selected by the first user on the first display area of the client system, the second business function component being associated with second application logic to access a second business application provided in the server system; and forming a relationship between the first and second business function components, wherein a visual representation of the user interface is created based on the displaying steps and the forming step (pg 242 for client systems and networks; pg 165 relationships

Art Unit: 2109

for business functions; p. 207 for operator function call.) The examiner interprets business functions as object oriented function calls for business applications.

Claim 11:

Access teaches a method of generation of a canonic representation of the visual representation; and generating an executable user interface code from the canonic representation, the user interface code being operable to access the first and second applications provided in the server system to retrieve desired data. (pg 21-43 for Visual Basic code). The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be compiled into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 12:

Access teaches a method of storing the user interface code in a repository in the server system, wherein the first and second business applications are different applications. (pg 242 for network system and pg 19, 31, and 37 on storing)

Claim 13:

Access teaches a method wherein the visual representation includes a third business function component that specifies a presentation format, the method

further comprising: storing the user interface code in a repository associated with the server system (pg 242 for network system and pg 19, 31, and 37 on storing); and receiving a request to access the user interface code from a second user, wherein the user interface code is executed in response to the request, the code being used to access the first and second applications provided in the server system to retrieve data desired by the second user, wherein the data retrieved for the second user is displayed on a second display area of the client system according to presentation format specified by the third business function component, the first and second display areas being associated with different client systems (pg 242 for network system and pg 21-43 for Visual Basic code). The examiner notes and sending and receiving requests to and from server and clients is an inherent property of network systems. The code can be written in a plurality of languages, including Visual Basic and XML.

Claim 14:

Access teaches a method of storing a first identifier for the first business function component in the client system in conjunction with the displaying-a-first-business-function-component step (pg 242 for network system and pg 19, 31, and 37 on storing); and storing a second identifier for the second business function component in the client system in conjunction with the displaying-a-second-business-function-component step, wherein the first and second identifiers are used subsequently at a runtime to access the first and second

application logics, respectively (pg 242 for network system and pg 19, 31, and 37 on storing).

Claim 15:

The method of claim 14, further comprising: generating a canonic representation of the visual representation; and generating a first executable user interface code from the canonic representation, the first user interface code being operable to access the first and second applications provided in the server system to retrieve desired data, wherein the first and second identifiers are inserted in the canonic representation. (pg 21-43 for Visual Basic Code). The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be complied into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 17:

The method of claim 15, further comprising: generating a second executable user interface code from the canonic representation, the second user interface code being operable to access the first and second applications provided in the server system to retrieve desired data, wherein the first code is compatible with a first platform and the second code is compatible with a second platform that is different from the first platform. (pg 21-43 for Visual Basic Code) The examiner

notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be complied into an executable code (<http://www.linuxjournal.com/article/0216>).

Claim 18:

Access teaches a method for associating an operator to the second business function component; and connecting an output port of the second business function component to an input port of the operator (pg 165 – relationship tool bar)

Claim 19:

Access teaches a method of generating a canonic representation of the visual representation; and generating an executable user interface code from the canonic representation, the first user interface code being operable to access the first and second applications provided in the server system to retrieve desired data, wherein the generated code does not require verification. (pg 21-43 for Visual Basic code) The examiner notes that it is well known in the art that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how source code can be complied into an executable code (<http://www.linuxjournal.com/article/0216>).

Art Unit: 2109

Claim 21:

Access teaches a method of claim 19, wherein the operator processes data received from the second business function component in such a way that the data remain consistent with the second application logic associated with the second business function component. (pg 165 – relationship tool bar); The examiner notes that business functions are interpreted as object oriented function calls for business applications.

Claim 22:

Access teaches a method wherein the first and second business function components are reusable application components. (pg 163–192 for creating an application and pg 314-327 for hyperlinks that can be used to run any program) The examiner interprets that a reusable application is any computer application. The examiner notes that business functions are interpreted as object oriented function calls for business applications.

Claim 28:

Access teaches a computer readable medium including a computer program, the computer program including: code for displaying a first business function component selected by a first user on a first display area of a client system, the first business function component being associated with first application logic to

Art Unit: 2109

access a first business application provided in a server system (p. 207 - The examiner notes that business functions are interpreted as object oriented function calls for business applications); code for displaying a second business function component selected by the first user on the first display area of the client system, the second business function component being associated with second application logic to access a second business application provided in the server system ; and code for forming a relationship between the first and second business function components, wherein a visual representation of the user interface is created based on the displaying steps and the forming step. (pg 21-43 for Visual Basic code, pg 242 for network system, pg 327 shows an example of a computer readable medium as a website builder, and pg 165 – relationship tool bar); The examiner notes that business functions are interpreted as object oriented function calls for business applications.

**Claim 29:**

Access teaches a code for generating a canonic representation of the visual representation; and code for generating an executable user interface code from the canonic representation, the user interface code being operable to access the first and second applications provided in the server system to retrieve desired data. (pg 21-43 for Visual Basic Code) The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry Ayers article on how



source code can be compiled into an executable code

(<http://www.linuxjournal.com/article/0216>).

Claim 30:

Access teaches a computer system, comprising: applications provided on a server system coupled to a client system (pg 242 for network system); and a computer readable medium including: code for displaying a first business function component selected by a first user on a first display area of the client system, the first business function component being associated with first application logic to access one or more business applications provided in a server system (pg 327 shows an example of a computer readable medium as a website builder with function buttons and p. 207 - The examiner notes that business functions are interpreted as object oriented function calls for business applications); code for displaying a second business function component selected by the first user on the first display area of the client system, the second business function component being associated with second application logic to access one or more business applications provided in the server system (pg 21-43 for Visual Basic code); and code for forming a relationship between the first and second business function components, wherein a visual representation of the user interface is created based on the displaying steps and the forming step. (pg 21-43 for Visual Basic code)

***Claim Rejections - 35 USC § 103***

5. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

6. The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

7.

Claim 8:

- Claim 8 is rejected under 35 U.S.C. 103(a) as being unpatentable over Access in view of Fitzloff et al ("Web Open Enterprise Portals" by Emily Fitzoff and Dana Gardner; from hereon will be known as Fitz').
- Access teaches one or more servers dedicated to the application layer and one or more Web servers dedicated to interface with a plurality of client systems (pg 242 for networks).
- Access fails to teach an enterprise portal.
- Fitz' teaches an enterprise portal (pg. 2, <http://www.infoworld.com/cgi-bin/displayStory.pl?features/990125eip.htm>).

Art Unit: 2109

- Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify Access and its GUI builder to include enterprise portals as taught by Fitz' because it allows GUI use in business applications.
- Both Access and Fitz' teach GUI's and web applications.

Claim 16:

- Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Access in view of Burkett et al (US 6476828; from hereon will be known as Burkett).
- Access teaches a canonic representation (Visual Basic)... wherein the first and second applications may be the same application or different applications. (pg 21-43)
- Access does not teach XML.
- Burkett teaches a canonic representation in XML. (Fig 3 A, B and Fig 4 A, B)
- Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify Access to use XML as a GUI builder as taught by Burkett because it allows the user to build a GUI in multiple languages including XML and Visual Basic. XML is inherently a robust language that can be used for multiple applications. (<http://en.wikipedia.org/wiki/XML>)
- Both Access and Burkett teach canonical representations and source code.

Claim 20:

Art Unit: 2109

- Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over Access in view of Santhanam et al (US 6247174; from hereon will be known as Sant').
- Access teaches a generated code (pg 21-43).
- Access fails to teach a requirement for performance tuning.
- Sant' teaches a generated code does not require a performance tuning. (col 6, lines 18-46 in reference to Fig 1)
- Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify Access to optimize the code so it does not have performance tuning as taught by Sant' because it prevents the user from modifying the generated code.
- Both Access and Sant' teach GUI's and source code.

Claims 24 – 27:

- Claims 24-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Access in view of Lennartsson et al (US 5446846; from hereon will be known as Len').
- With regards to claims 24 - 27, Access teaches a means for displaying a first business function component selected by a first user on a first display area of a client system, the first business function component being associated with first application logic to access a first business application provided in a server system (pg 242 for network system and pg 207 for business functions; The examiner notes that network systems are synonymous with server and client

systems and business functions are interpreted as object oriented function calls for business applications. Also, the examiner interprets distributed computer system as a network system.); means for displaying a second business function component selected by the first user on the first display area of the client system, the second business function component being associated with second application logic to access a second business application provided in the server system (pg 242 for network system and pg 207 for business functions; The examiner notes that network systems are synonymous with server and client systems and business functions are interpreted as object oriented function calls for business applications); and means for forming a relationship between the first and second business function components, wherein a visual representation of the user interface is created based on the displaying steps and the forming step (pg 165 – relationship tool bar). The examiner notes that business functions are interpreted as object oriented function calls for business applications.

- Access also teaches the limitations of claim 25; with respect to the claim, Access teaches a means for generating a canonic representation of the visual representation; and means for generating an executable user interface code from the canonic representation, the user interface code being operable to access the first and second applications provided in the server system to retrieve desired data. (pg 21-43) The examiner notes that it is inherent that Visual Basic is a form of source code that can be can be compiled into executable code, like any other source code. See Larry

Ayers article on how source code can be compiled into an executable code (<http://www.linuxjournal.com/article/0216>).

- o Access also teaches the limitations of claim 26; with respect to the claim, Access teaches a system of further comprising: means for storing the user interface code in a repository in the server system. (pg 242 for network system and pg 19, 31, and 37 on storing)
  - o Access also teaches the limitations of claim 27; with respect to the claim, Access teaches a visual representation includes a third business function component that specifies a presentation format, the system further comprising: means for storing the user interface code in a repository associated with the server system (pg 242 for network system, pg 19, 31, and 37 on storing and pg 319 for a webpage or visual representation with function buttons); and means for receiving a request to access the user interface code from a second user, wherein the user interface code is executed in response to the request, the code being used to access the first and second applications provided in the server system to retrieve data desired by the second user, wherein the data retrieved for the second user is displayed on a second display area of the client system according to presentation format specified by the third business function component, the first and second display areas being associated with different client systems. (Pg 21-43 for Visual Basic code and PG 242 for network system)
- The examiner notes and sending and receiving requests to and from

server and clients is an inherent property of network systems. The examiner also notes that code written in any robust language can program this, including Visual Basic and XML.

- With regard to claim 24, Access does not teach a distributed computer system.
- Len' teaches a distributed computer system "arrangement with master units and slave units which communicate with one another via a serial bus connection and in which the slave units can be connected to the connection via connecting devices and on their respective connection can be assigned identifications in the system." (col 1, lines 5-11)
- Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify Access to into a distributed computing system as taught by Len' because it allows GUI applications to be performed in a non-centralized networking environment.
- Both Access and Len' teach networks.

Claim 23:

- Claim 23 is rejected under 35 U.S.C. 103(a) as being unpatentable over Access in view of Ching et al (US 6407761; from hereon will be known as Ching') and in further view of Babutzka et al (US 6898794; from hereon will be known as Bab').
- Access teaches reusable application components. (pg 314-327 for hyperlinks that can be used to run any program). The examiner interprets that a reusable application is any computer application.

- Access fails to teach a business application program interface (BAPI) and a remote function call (RFC).
- Ching include a business application program interface (BAPI). (Fig 5 which describes the parameters of the BAPI interface)
- Both Access and Ching fail to teach RFC's.
- Bab' teaches a RFC (The examiner notes that in the abstract the inventor Bab'8 improves "remote function calls").
- Accordingly, it would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify Access to include BAPI's and RFC's as taught by Ching and Bab' respectively because it allows GUI applications to be used with business applications in a remote network environment. The examiner notes that it is inherent that RFC's are used in ABAP or advanced business applications. Since a BAPI is a business application interface (or an user interface for a plurality of business applications), ABAP (or advanced business applications programming) languages can be used in BAPI's.  
(<http://en.wikipedia.org/wiki/ABAP>,  
[http://en.wikipedia.org/wiki/Remote function call](http://en.wikipedia.org/wiki/Remote_function_call)).
- Access, Ching, and Bab' teach applications that can be run on GUI's.

### ***Conclusion***

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.



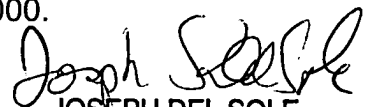
Art Unit: 2109

- The examiner also issues a supplemental notes that Wenzel et al (US 7043693; Fig 23A-D) also shows a diagram relationship)
- A supplemental discussion on logging into networks to use a computer application (i.e. MS Frontpage) is discussed in Verio –“Logging into your Server from FrontPage” at [http://www.verio.com/support/documents/view\\_article.cfm?doc\\_id=3703](http://www.verio.com/support/documents/view_article.cfm?doc_id=3703).

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Soumya (Ronnie) Dasgupta whose telephone number is 571-270-7432. The examiner can normally be reached on Monday through Friday 7:30 am to 5:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Joseph Del Sole can be reached on 571-272-1130. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

  
JOSEPH DEL SOLE  
SUPERVISORY PATENT EXAMINER  
3/27/07